Bilateral and Trilateral Adaptive Support Weights in Stereo Vision

Ryan Beethe

Student, Colorado School of Mines

rbeethe@mines.edu

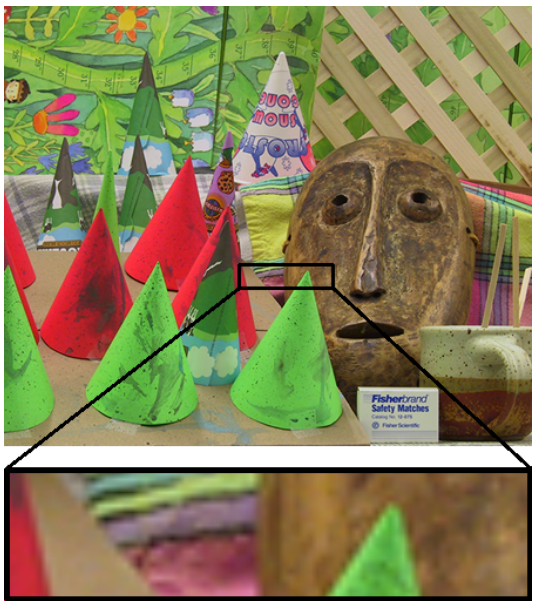
April 7, 2016

Overview

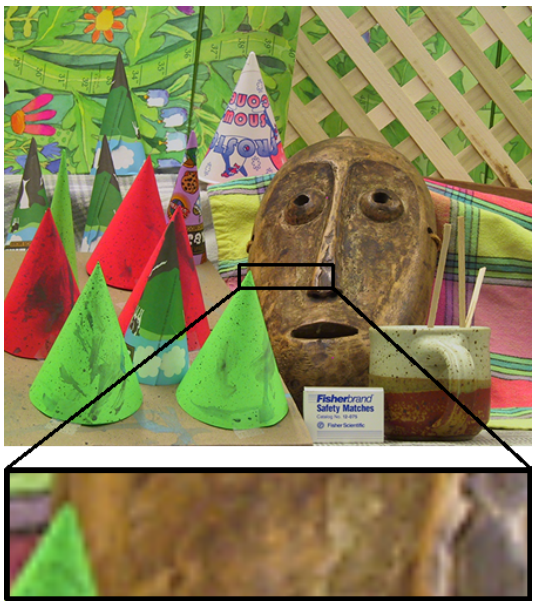
Stereo Matching

Stereo Image Example [3]

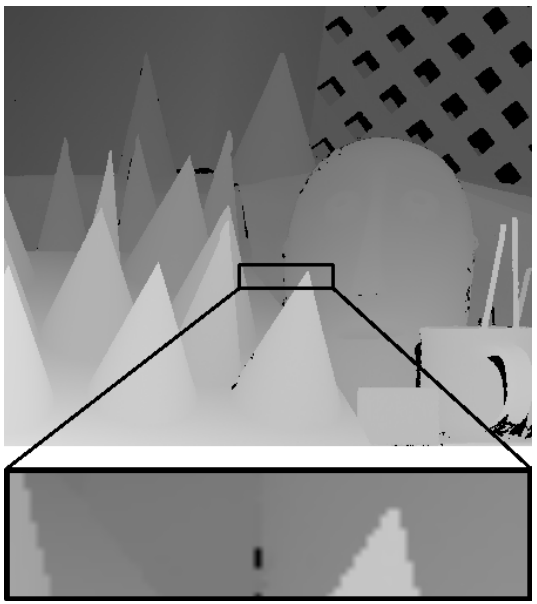
Reference Image (left)



Target Image (right)



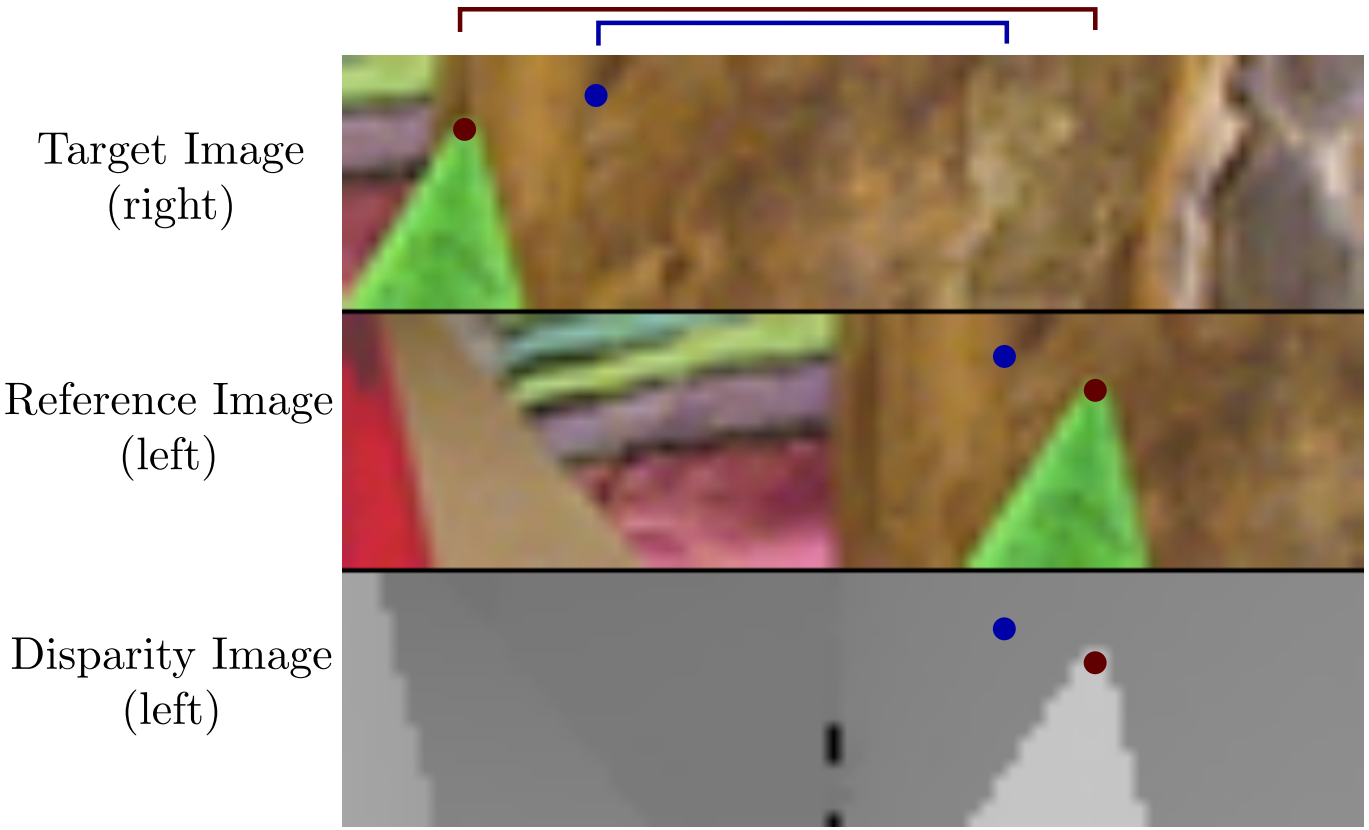
Disparity Image (left)



We are going to use corresponding regions of these three images to show how a disparity image is generated.

Stereo Image Example

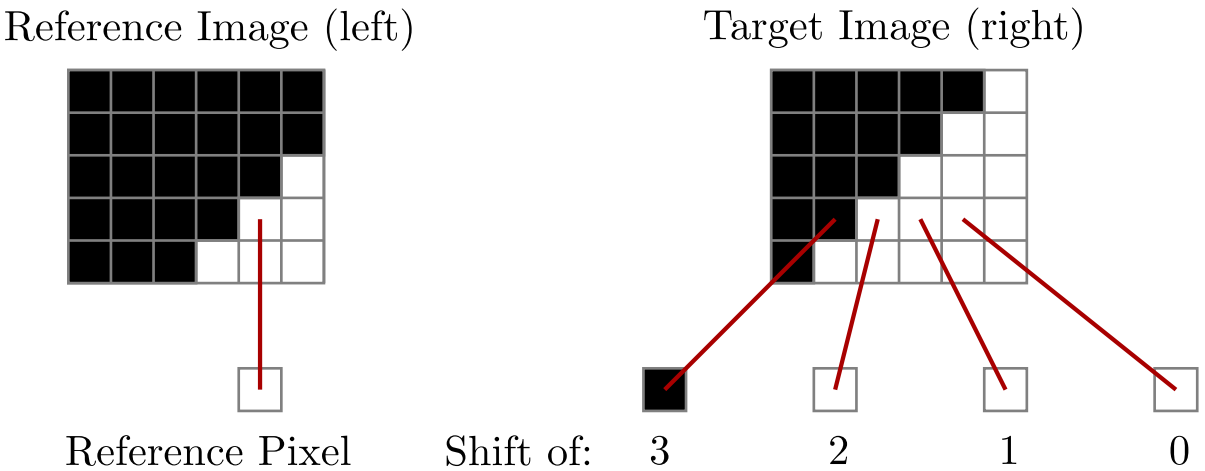
How Disparity Image is Generated



The shift between red dots is greater than the shift between blue dots, so the disparity image is brighter under the red dot.

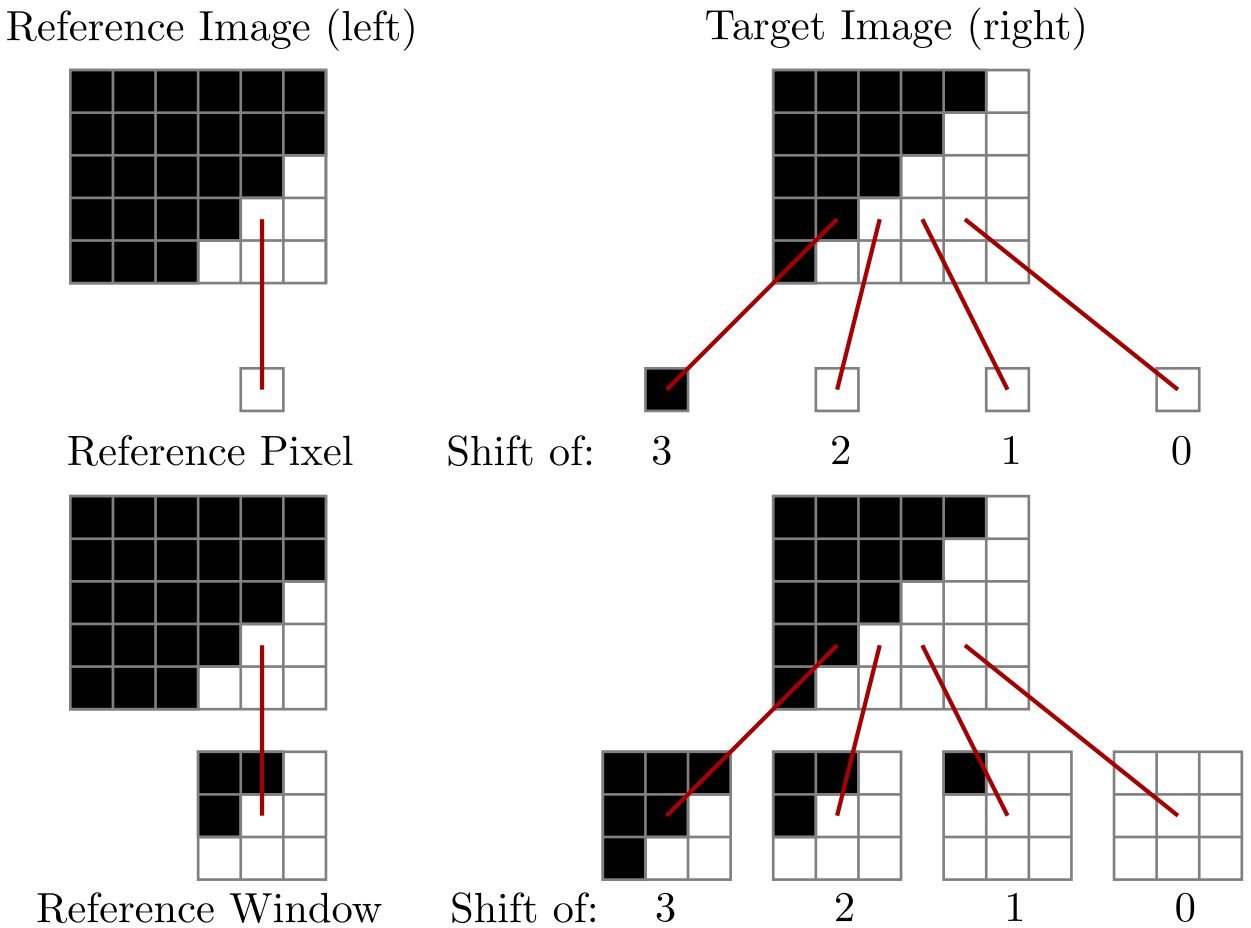
Pixel Matching

Ambiguity issues



Block Matching

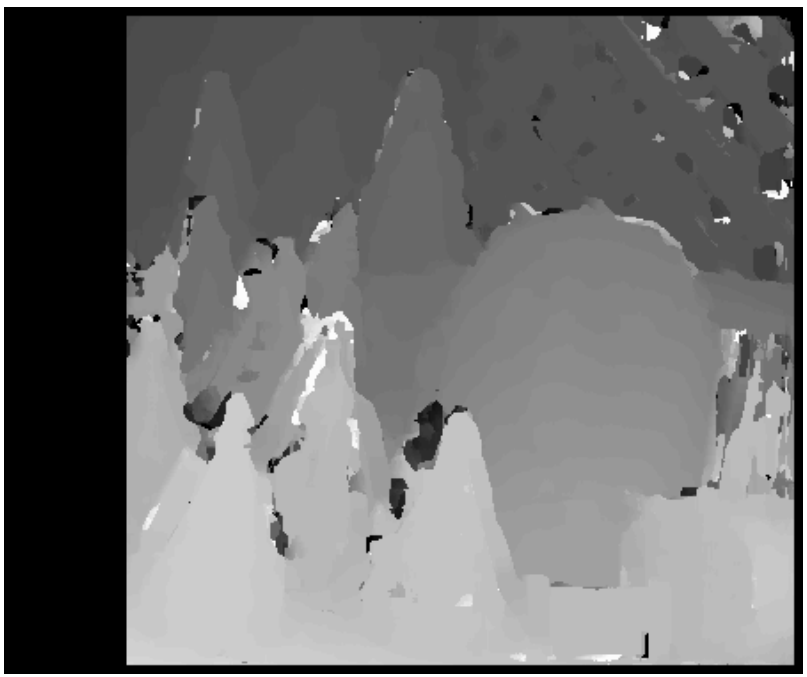
Blocks of Pixels Reduce Ambiguity



Block Matching

Left Image

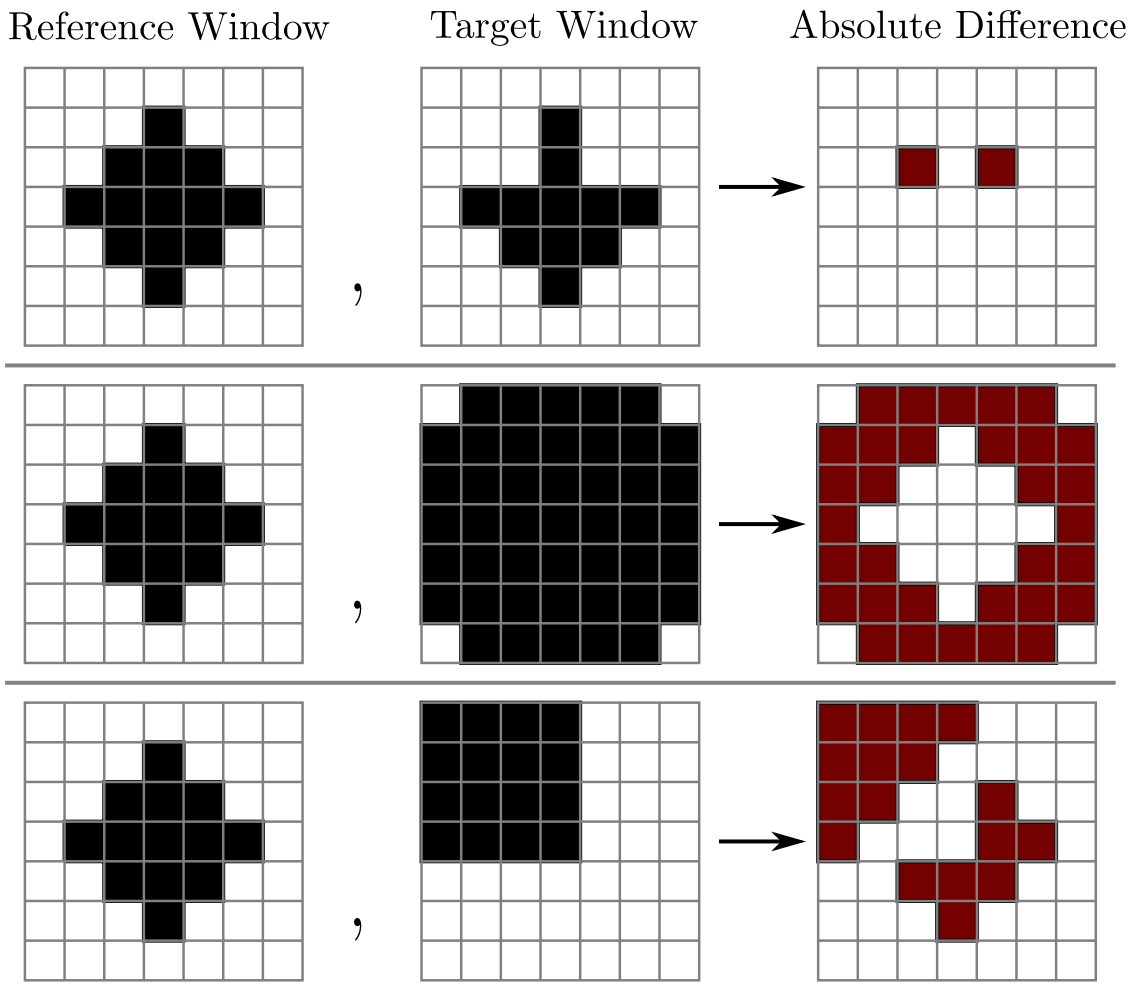
Disparity Image



Raw block matching is very noisy.

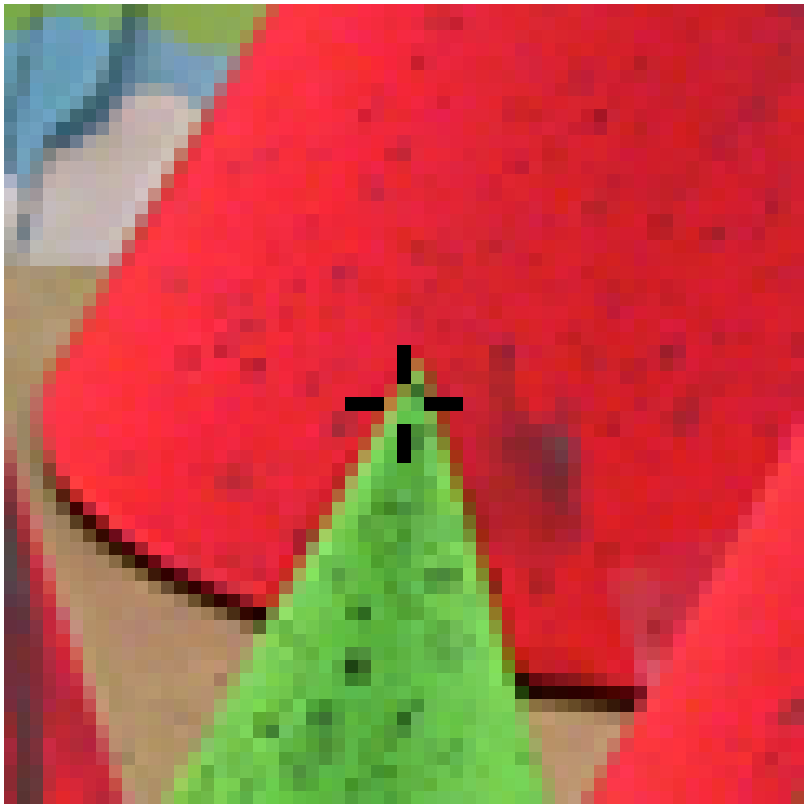
Matching Cost, Cost Aggregation

A common "matching cost function" is the absolute difference between corresponding pixels. A simple "cost aggregation" is the sum of absolute differences (SAD) between two windows.



SAD Limitation

What is wrong with using SAD on this reference window?



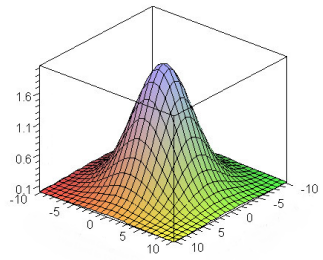
There are more background pixels in the window than cone pixels. As a result, SAD will yield a lower aggregate matching cost at the background disparity than at the (correct) cone disparity.

Bilateral Filtering

Bilateral Explanation

Edge-Preserving Version on Gaussian

Gaussian kernel (regardless of pixel values)



Bilateral "kernel" at a noisy step [4]

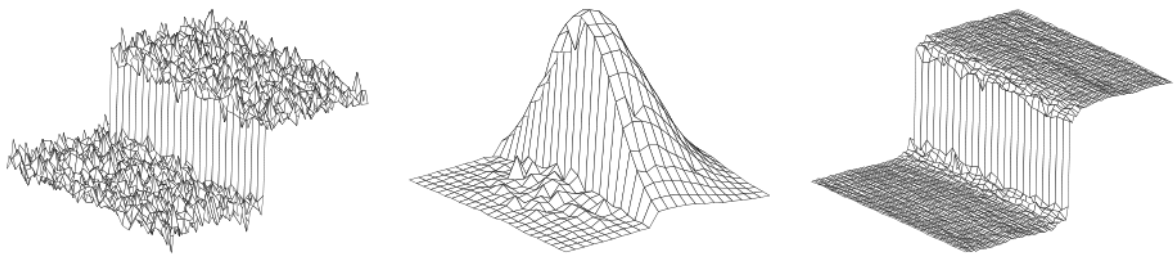


Figure 1: (a) A 100-gray-level step perturbed by Gaussian noise with $\sigma = 10$ gray levels. (b) Combined similarity weights $c(\xi, \mathbf{x})s(\mathbf{f}(\xi), \mathbf{f}(\mathbf{x}))$ for a 23×23 neighborhood centered two pixels to the right of the step in (a). The range component effectively suppresses the pixels on the dark side. (c) The step in (a) after bilateral filtering with $\sigma_r = 50$ gray levels and $\sigma_d = 5$ pixels.

Bilateral Example

Original Image

Bilaterally Smoothed Image

Notice that the image textures have been blurred, but the edges remain sharp. The bilateral filter does not smooth across sharp edges.

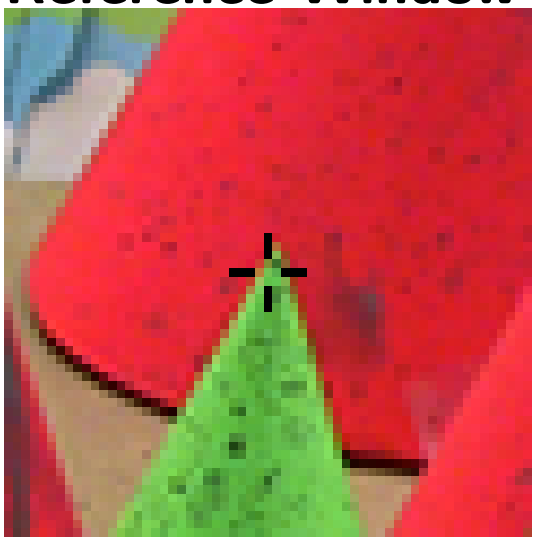
Adaptive Support Weight

Introduction

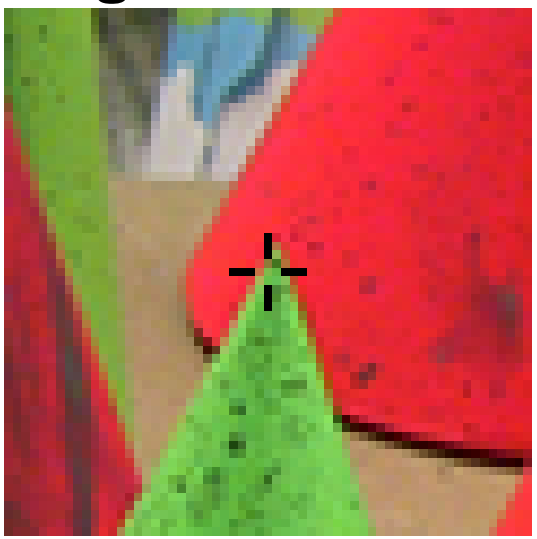
Adaptive support weight uses a bilateral "kernel" as the weighting function in the cost aggregation calculation.

Consider the image windows below, one Reference window and two possible matching target windows:

Reference Window



Target Window 1

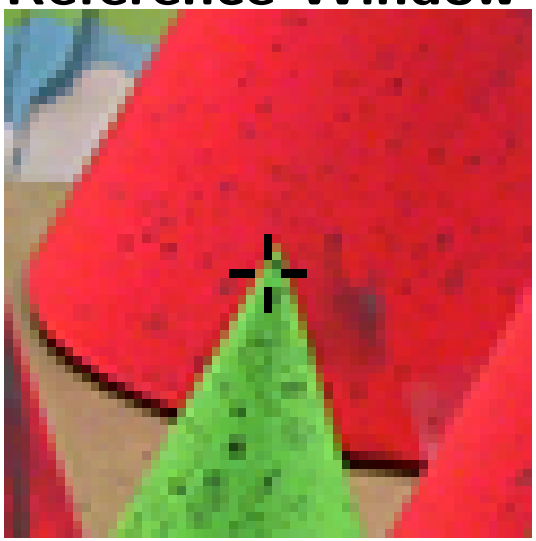


Target Window 2

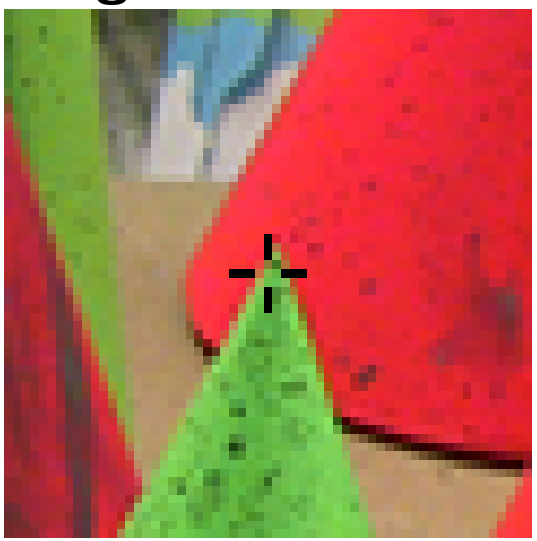
Adaptive Support Weight

Individual Support Weights

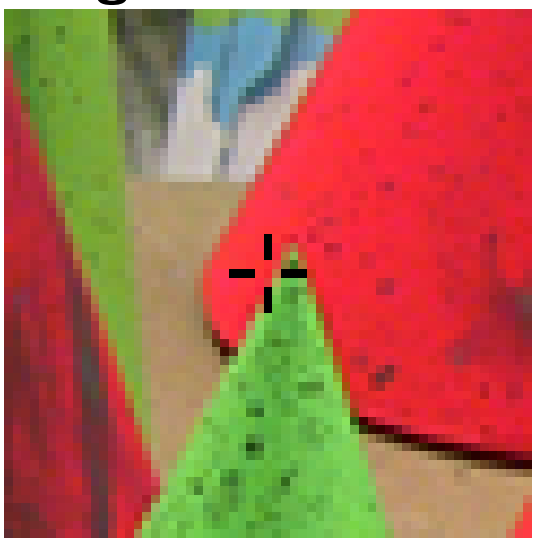
Reference Window



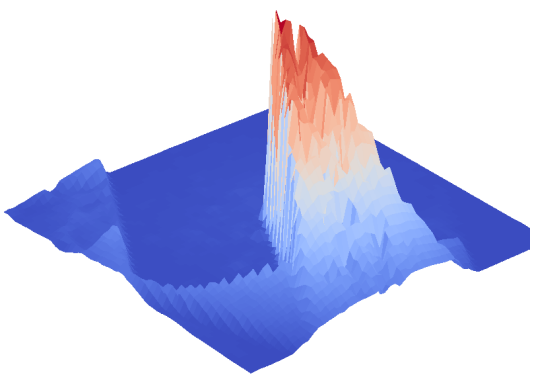
Target Window 1

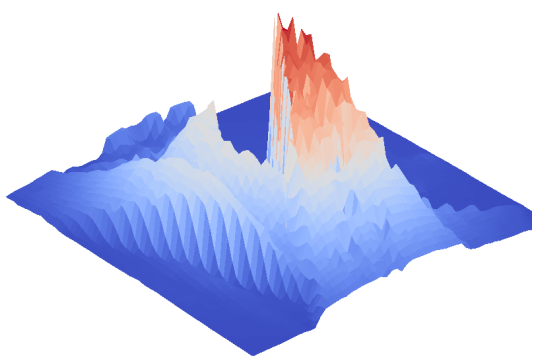


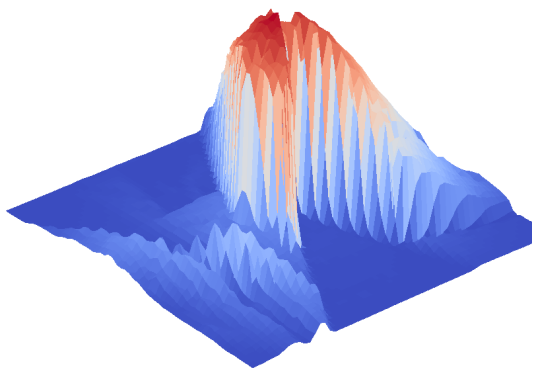
Target Window 2



Individual Window Support Weights





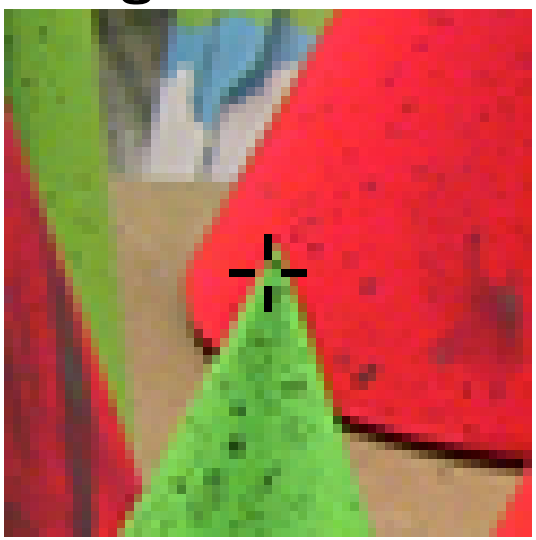


Adaptive Support Weight

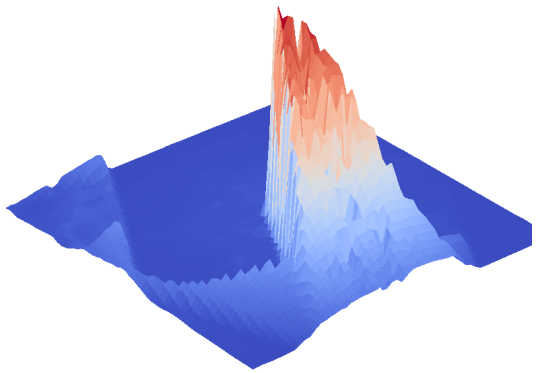
Combined Support Weights

Reference Window

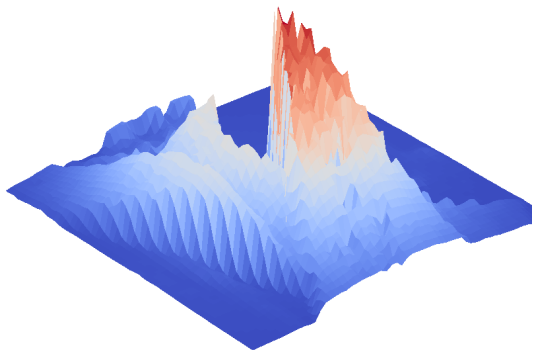
Target Window 1



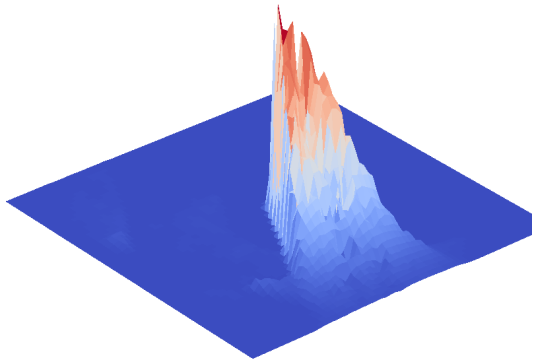
Reference Window Support Weight



Target Window 1 Support Weight



Combined Support Weight



Support weight is given to the area of the green cone, an area which will match well.

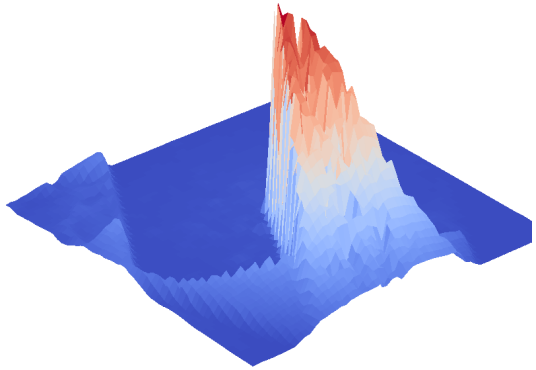
Adaptive Support Weight

Combined Support Weights, cont.

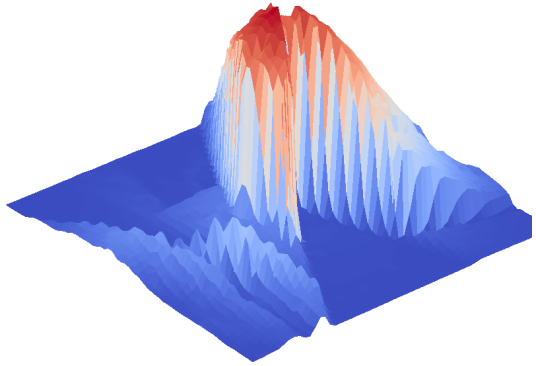
Reference Window

Target Window 2

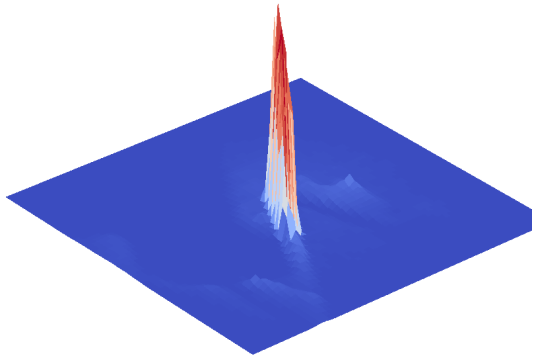
Reference Window Support Weight



Target Window 2 Support Weight



Combined Support Weight



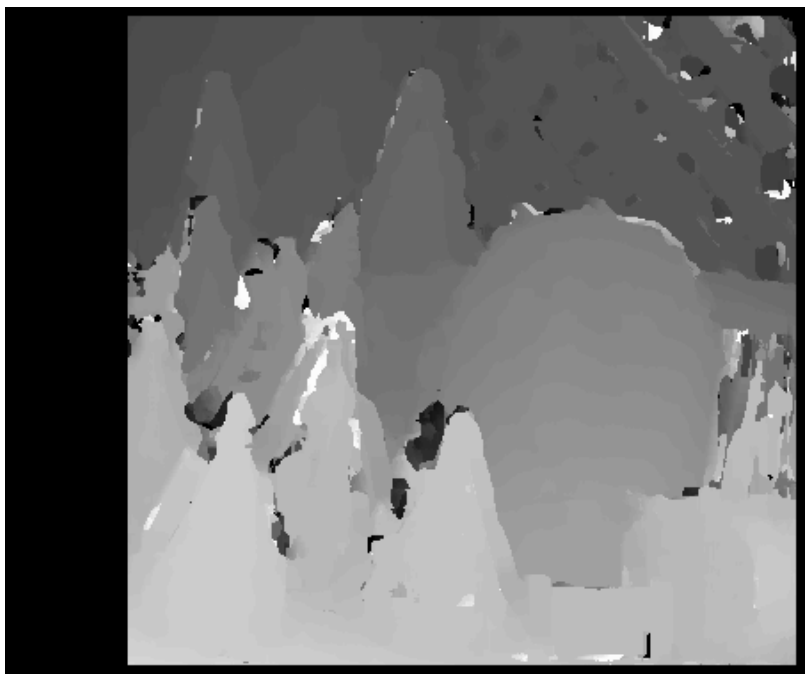
Support weight is given to areas which are green in Reference but red in Target, which will match poorly.

Adaptive Support Weight

Improved Stereo Output

ASW is a huge improvement. There is far less noise. Notice that the tips of the cones in particular are much sharper.

Raw BM Output



ASW Output[5]

Other ASW Methods

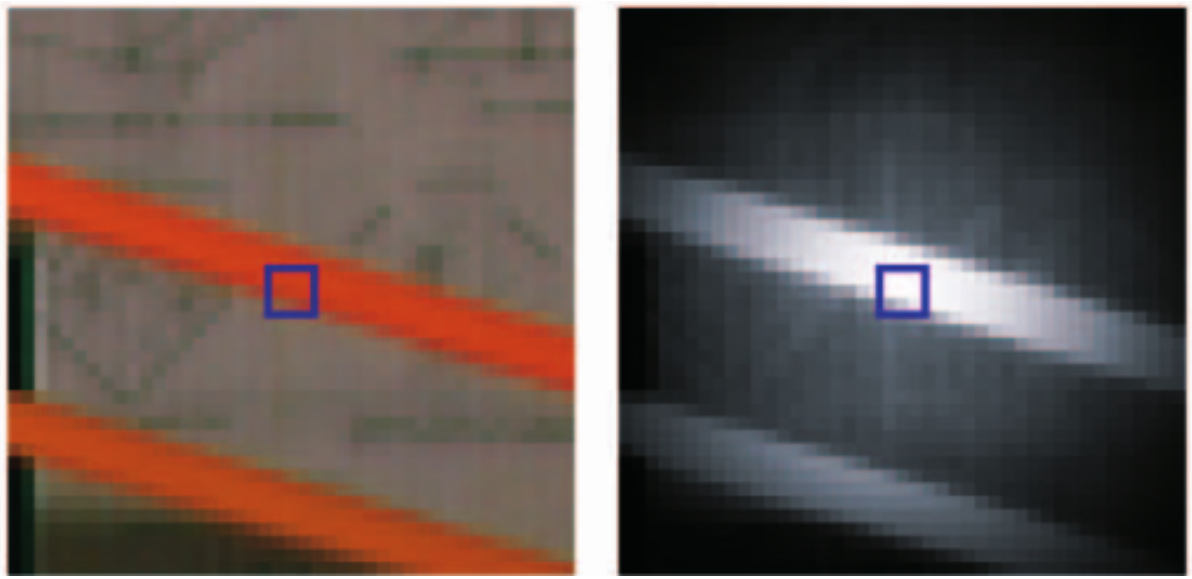
Many alternative methods for adaptive support weight have been proposed based on the original bilateral-based ASW implementation. Modifications include:

- Removing spacial component of bilateral filter
- Replacing bilateral with guided filter
- Replacing bilateral filter with some other filter
- Approximating the bilateral filter
- Calculating support weight for only the reference image

For more discussion, see [2].

Bilateral ASW Limitations

Should the lower red area really be weighted so high? It's not clear that those two objects should be at the same depth.



The bilateral filter implicitly assumes similarly colored, disconnected-but-nearby objects are the same object.

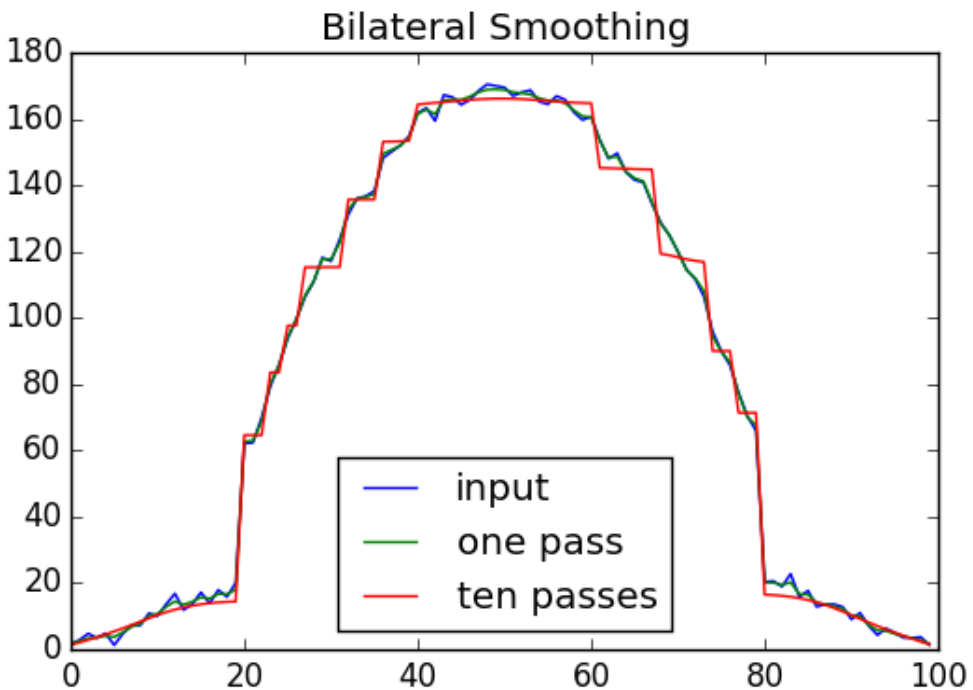
Image from [5].

Trilateral Filtering

Trilateral Explanation

Trilateral filter, in terms of Bilateral

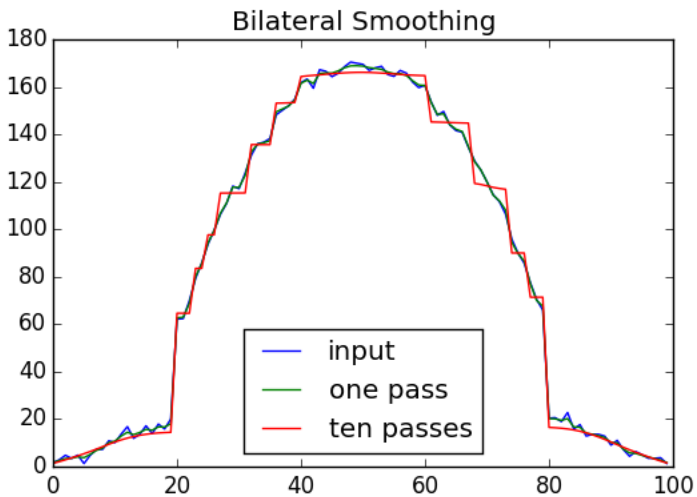
The bilateral filter smooths an image, according to regions of similar color. It settles into a piecewise-flat solution, with steps approximately equal to the color σ.

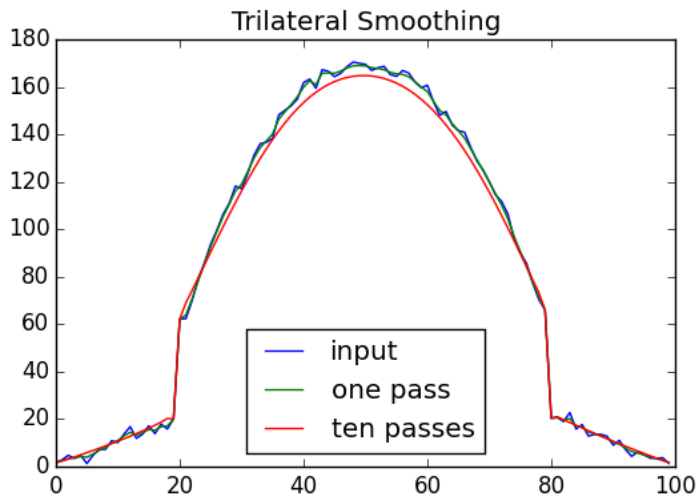


Trilateral Explanation

Trilateral filter, in terms of Bilateral, cont.

The trilateral filter applies a bilateral filter to an image gradient. The result is a piecewise-flat gradient, and therefore a piecewise smooth image.

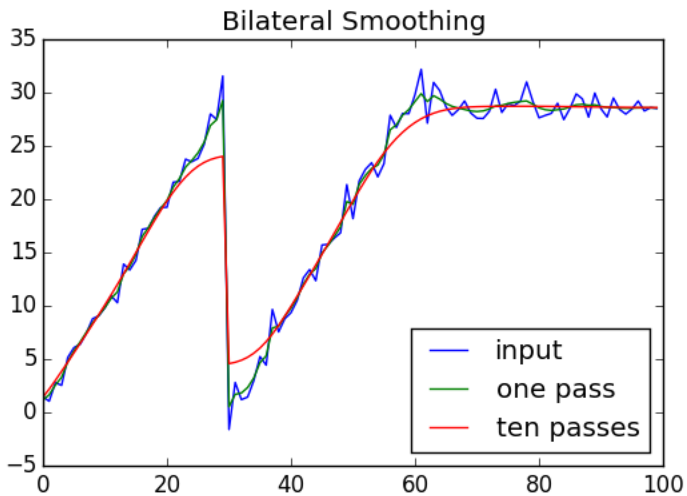


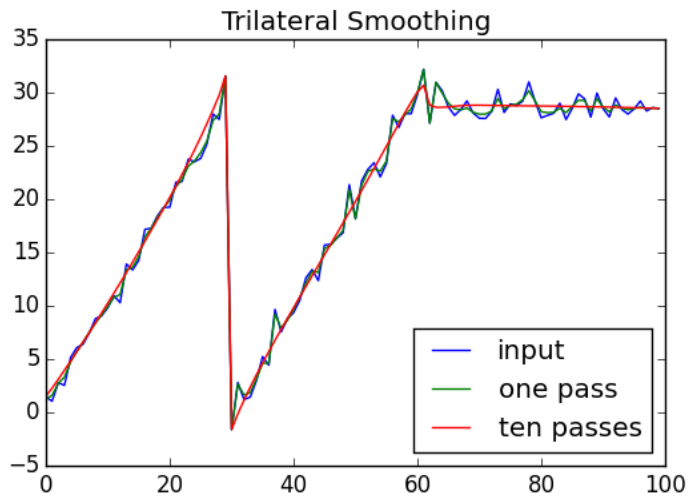


Trilateral Explanation

Cut-off Term

In order to not smooth across boundaries, the trilateral filter only smooths across a connected-component region of similar gradient. The connected-component region of similar gradient is called a "neighborhood" function, but it is really just a binary mask.

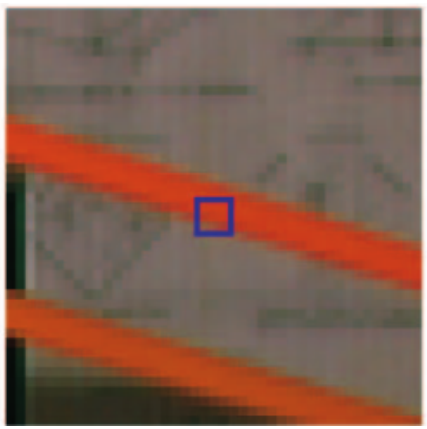


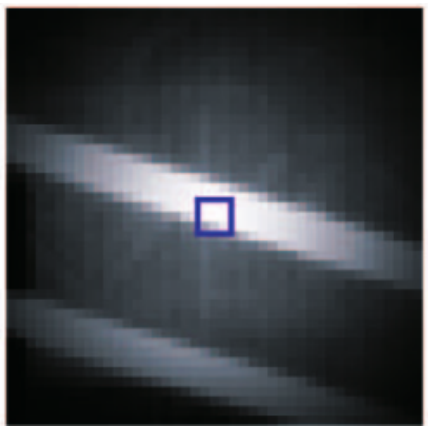


Trilateral-based ASW

Trilateral-based ASW improves the original ASW by introducing an additional step to reduce support weights of pixels across color boundaries [1].

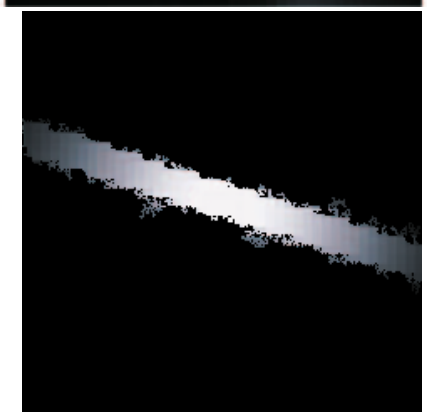
Original Window





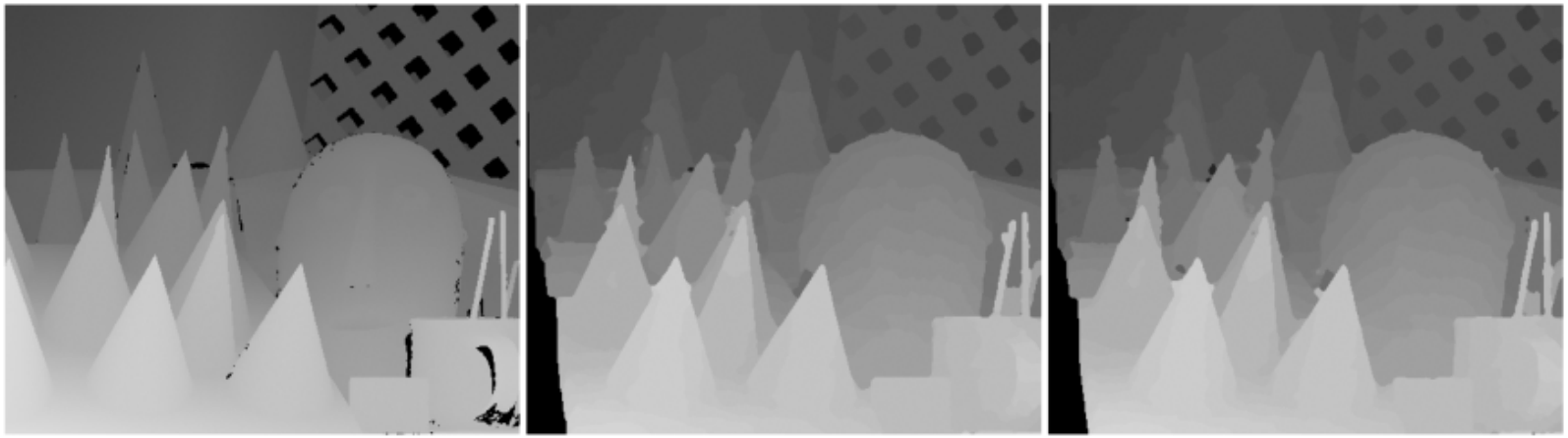
Bilateral support weight

Additional support weight term



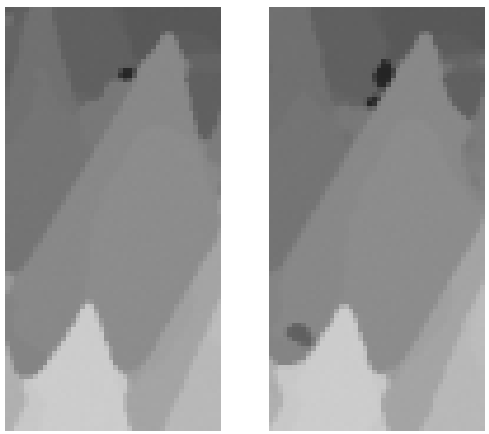
Combined support weight

Trilateral-based ASW



ground truth | trilateral filter (test 4) | bilateral filter (test 5)

Notice the trilateral-based ASW (left) outpreforms bilateral-based ASW (right) near edges:



Images adapted from [1]

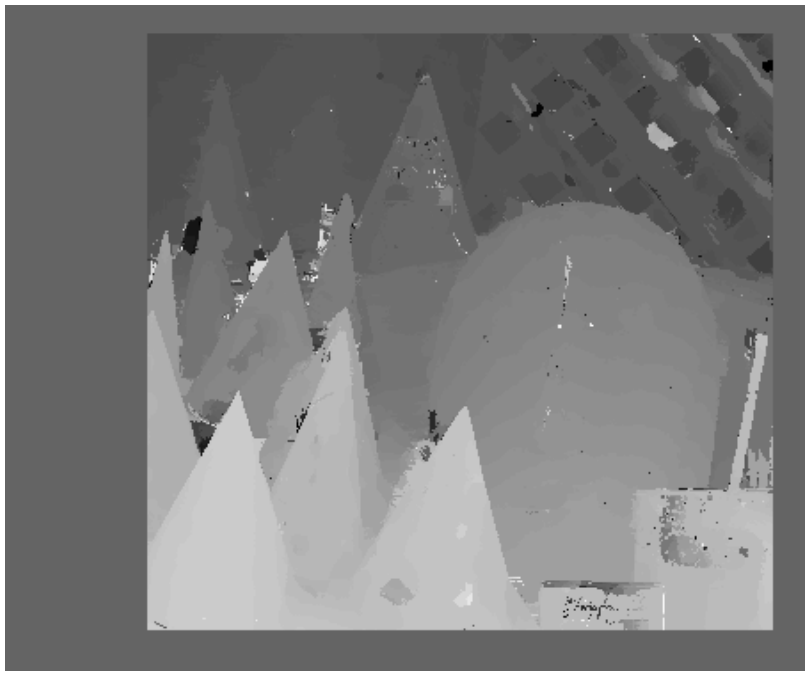
Trilateral Performance

Image	Algorithm	nonocc	all	disc
Tsukuba	Semi-Global BM	3.26	3.96	12.8
	Trilateral ASW	1.65	1.96	5.90
	Bilateral ASW	1.38	1.85	6.90
Venus	Semi-Global BM	1.00	1.57	11.3
	Trilateral ASW	0.14	0.31	1.51
	Bilateral ASW	0.71	1.19	6.13
Teddy	Semi-Global BM	6.02	12.2	16.3
	Trilateral ASW	6.25	11.8	15.1
	Bilateral ASW	7.88	13.3	18.6
Cones	Semi-Global BM	3.06	9.75	8.90
	Trilateral ASW	2.49	8.32	7.02
	Bilateral ASW	3.97	9.79	8.26

The trilateral filter does especially well near discontinuities in the disparity image, abbreviated as “disc”.

ASW in Practice

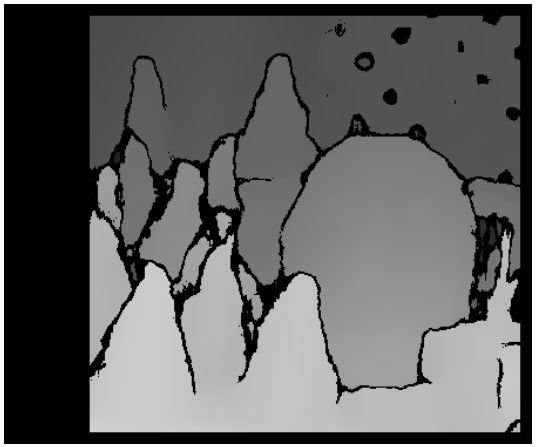
GPU Adaptive Support Weight



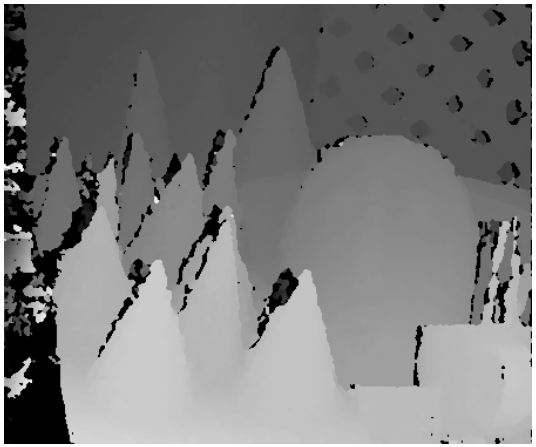
Preliminary GPU ASW
Runs as fast as 94 ms
(Quadro K6000)

ASW (from paper)[5]
Takes almost 60 seconds
(CPU-only).

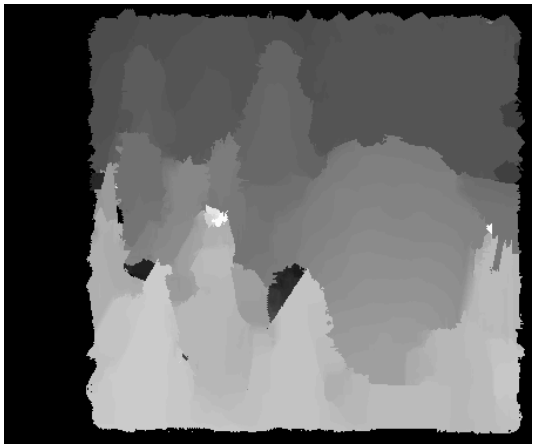
OpenCV Options



Block Matching
Has speckle filter, uniqueness checks
Gray-scale only
Has GPU version



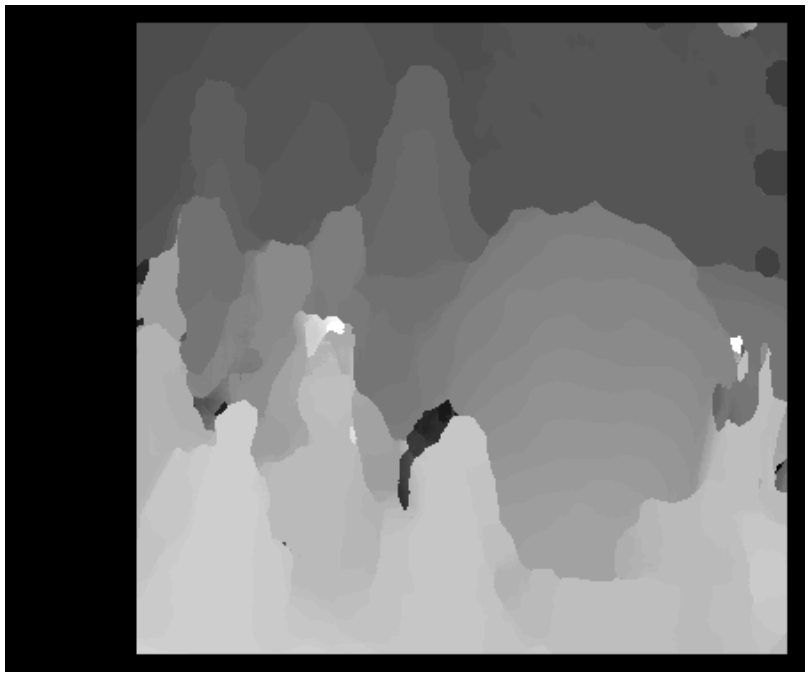
Semi-Global Block Matching
Global Reasoning
Smoother Output
No OpenCV GPU version



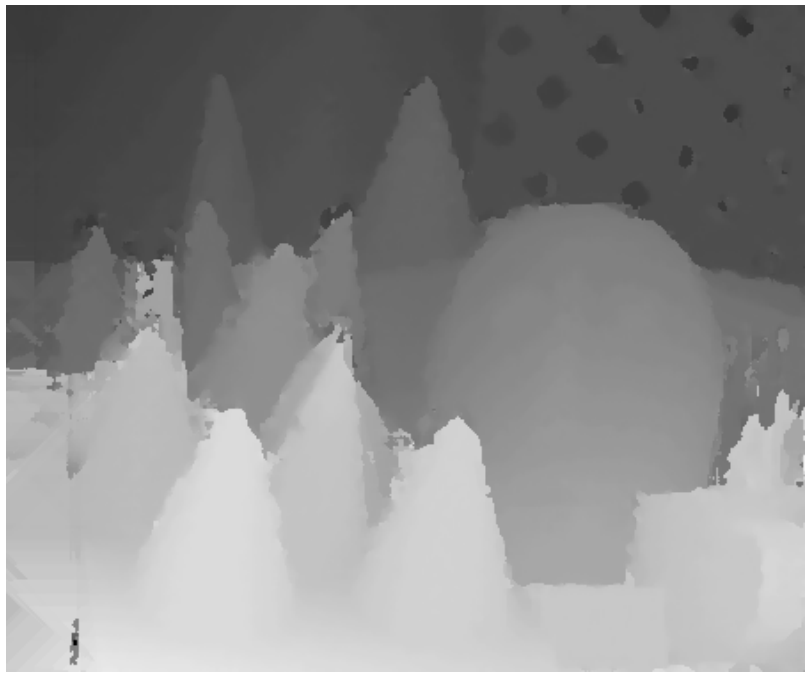
Disparity Bilateral Filter
Post-processing for disparity images, only implemented on GPU

Belief Propagation and Constant-Space Belief Propagation not shown

Nvidia VisionWorks



Block Matching
No Speckle Filter
No uniqueness checks
Similar to OpenCV GPUBM, but faster and cleaner



Semi-Global Block Matching
Slightly less accurate
Very Fast
Includes post-processing

Timing Comparison

Tests ran on 450x375 "cones" images

Algorithm	Jetson TK1	Jetson TX1	i7 + Quadro K6000
CPU BM	29ms	29ms	12ms
GPU BM	18ms	9.5ms	2.2ms
GPU DBF	64ms	21ms	10ms
CV SGBM	870ms	990ms	99ms
VX BM	13ms	5.7ms	2.3ms
VX SGBM	65ms	42ms	5.1ms
GPU ASW	8,900ms	6,800ms	94ms
CPU ASW	190,000ms	200,000ms	73,000ms

Source code for test programs can be found at:
`https://github.com/rdbeethe/gtc_tests`

Conclusion

ASW Pros:

- Excellent edge accuracy
- High disparity detail
- Less noise
- Parallelizes well

ASW Cons:

- Much higher computational complexity
- No completed open-source GPU version

My GPU-accelerated ASW implementation can be found at:
`https://github.com/rdbeethe/asw`

References I


Dongming Chen, Mohsen Ardabilian, and Liming Chen.
A novel trilateral filter based adaptive support weight method for stereo matching.
IEEE Transactions on Circuits and Systems for Video Technology, 25:730–743, 2014.


Asmaa Hosni, Michael Bleyer, and Margrit Gelautz.
Secrets of adaptive support-weight for local stereo matching.
Computer Vision and Image Understanding, 117:620–632, 2013.


Daniel Scharstein and Richard Szeliski.
A taxonomy and evaluation of dense two-frame stereo correspondence algorithms.
International Journal of Computer Vision, 2002.

References II


C. Tomasi and R. Manduchi.
Bilateral filtering for gray and color images.
IEEE International Conference on Computer Vision, 1998.


Kuk-Jin Yoon and In So Kweon.
Adaptive support-weight approach for correspondence search.
IEEE Transactions on Pattern Analysis and Machine Intelligence, 28:650–656, 2006.

Questions?